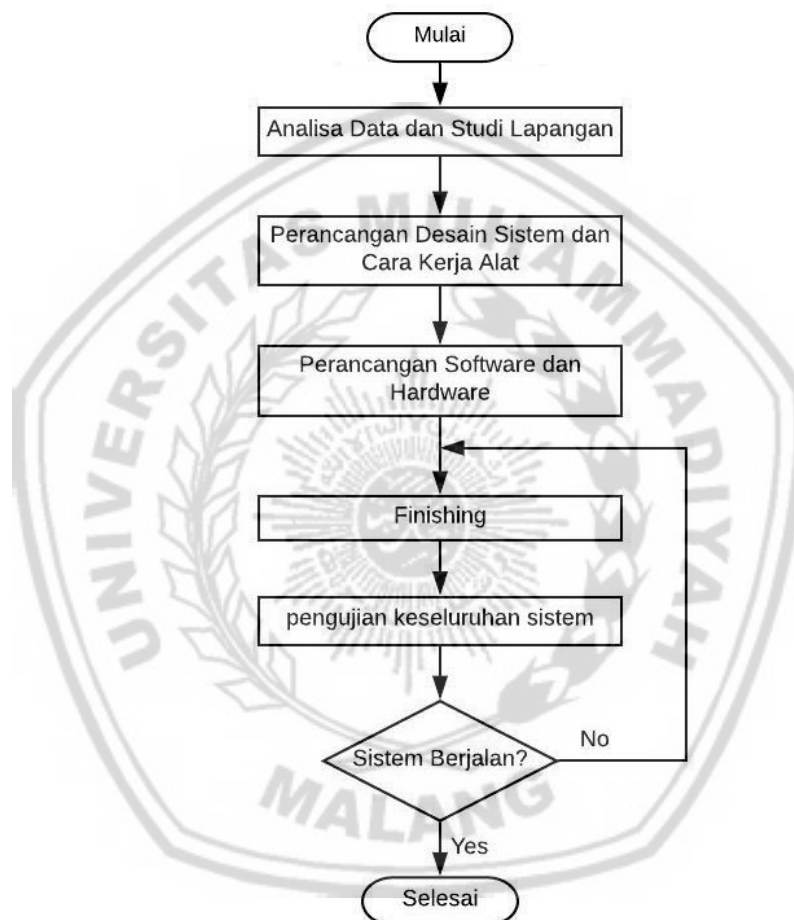


### BAB III METODOLOGI PENELITIAN

Pada perancangan sistem ini akan dijelaskan bagaimana dan darimana akan memulai proses pembuatan sistem pendeteksi tingkat dehidrasi tubuh berdasarkan warna dan kadar amonia urin terintegrasi aplikasi *smartphone*. Tahapan-tahapan dalam pembuatan alat memiliki alur sebagai berikut:



**Gambar 3.1** Flowchart Diagram Alir Penelitian

#### 3.1 Analisa Data dan Studi Lapangan

Pada tahap ini, teknologi yang sudah ada yaitu *Alat Deteksi Dehidrasi Menggunakan LED dan Fotodiode Melalui Warna Urin* dianalisa sehingga dapat diketahui bagaimana sistem teknologi itu berjalan dan apakah kekurangan dari sistem tersebut. Selanjutnya studi lapangan dilakukan ke laboratorium kesehatan untuk meminta data hasil uji laboratorium mengenai tingkat dehidrasi tubuh

manusia berdasarkan urin dimaksudkan untuk mendapat data yang kredibel berkenaan dengan tingkat dehidrasi tubuh

### 3.2 Perancangan Sistem

Perancangan sistem dibuat berdasarkan kebutuhan peneliti untuk lebih mudah memetakan dan menganalisa sistem yang akan dikerjakan. Sistem yang akan dikerjakan pada penelitian ini yaitu perancangan sistem perangkat lunak dan perancangan perangkat keras. Dalam tahap perancangan sistem perangkat lunak, bagaimana algoritma program mikrokontroler dan software aplikasi android dari alat ini dapat diketahui sesuai dengan mekanisme kerja alat secara keseluruhan. Tahap-tahap perancangan *software* mikrokontroler dan *software* aplikasi android dilakukan dengan cara:

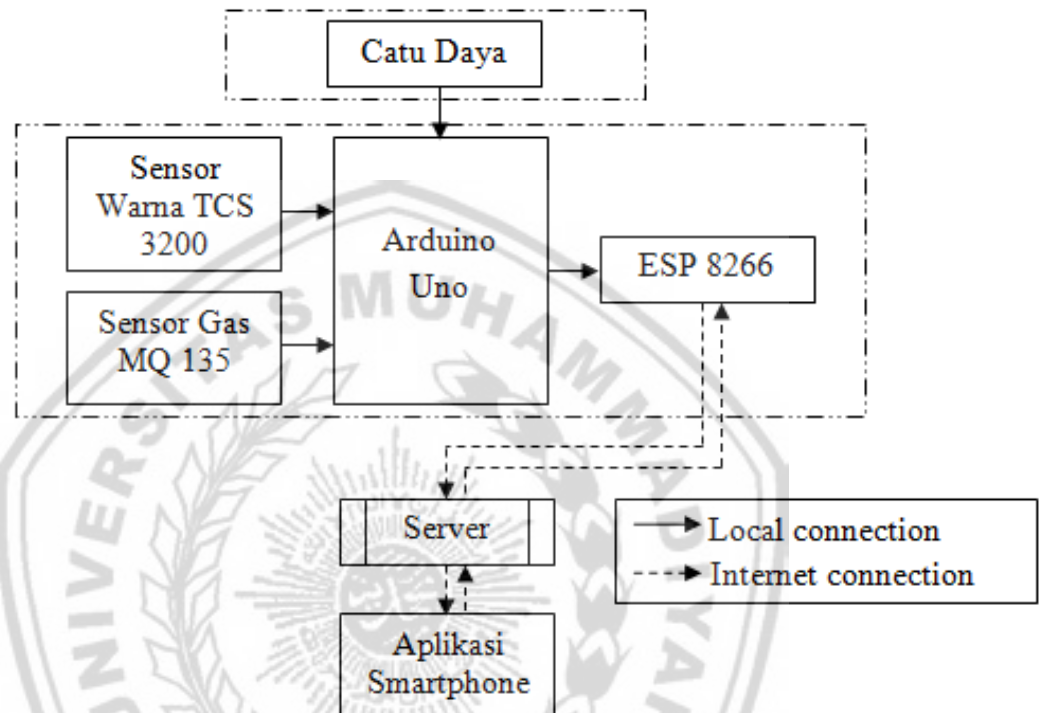
1. Outlining, pada tahap ini menjelaskan gambaran umum dari software mikrokontroler dan aplikasi android sebagai media informasi tingkat dehidrasi.
2. Storyboarding, alur tampilan atau *user interface* dari aplikasi android sebagai media informasi hasil pendeteksi tingkat dehidrasi tubuh seperti :
  - a. Scene 1 – merupakan tampilan awal saat pertama kali membuka aplikasi android berupa halaman login.
  - b. Scene 2 – Tampilan menu utama, dalam scene ini terdapat beberapa scene lagi yang menjelaskan bagian dari proses menu utama seperti tombol “Ambil Data” dan halaman data hasil deteksi tingkat dehidrasi urin yang disajikan dalam tampilan tingkatan dehidrasi dan saran pengobatan.
3. Flowcharting, tahap ini menjelaskan proses dan prosedur yang terjadi pada aplikasi android atau *user experience* dengan menggunakan simbol-simbol flowchart untuk mengetahui alur dari aplikasi.
4. Hierarki Model, konsep navigasi ini dimulai dari yang menjadi halaman utama atau halaman awal, dari halaman tersebut dapat dibuat beberapa cabang kehalaman level 1, dari tiap halaman level 1 dapat dikembangkan menjadi beberapa cabang lagi.

Pada perancangan perangkat keras meliputi bagian catu daya *input*, *kontroller*, *input output*, dan *output*. Pada bagian input terdiri dari Sensor warna

TCS 3200 dan Sensor gas MQ 135, dan sistem android. Pada bagian kontroller menggunakan arduino uno yang berfungsi sebagai pusat pengolahan data seluruh sistem. Dan pada bagian output terdiri dari modul ESP 8266.

### 3.2.1 Desain Sistem Mikrokontroller

Proses desain sistem dilakukan seperti pada blok diagram dibawah ini :



**Gambar 3.2** Blok Diagram Sistem

Desain sistem pendeteksi tingkat dehidrasi tubuh berdasarkan warna dan kadar amonia urin terintegrasi aplikasi smartphone ini dibagi 5 subsistem yaitu bagian catu daya, input, kontroler, input output dan output . Pada bagian catu daya terdiri dari ADC dengan output 5 volt sebagai penyuplai daya.

Pada bagian kontroler menggunakan arduino uno yang berfungsi sebagai pusat pengolahan data. Sistem membutuhkan 12 pin arduino uno yang dibagi 5 pin untuk input sensor warna TCS 3200 pada pin 4,5,6,7, dan output pin 3. Input output sensor gas MQ 135 pada pin A0, VCC dan GND. Selanjutnya 3 pin untuk input output modul ESP 8266 pada pin 9 (TX), 8 (RX) dan GND.

Pada bagian input sistem terdapat Sensor warna TCS 3200 berfungsi untuk mendeteksi warna urin yang berada dalam wadah urin. Sensor gas MQ 135 berfungsi untuk mengetahui kadar amonia yang terkandung pada urin yang

bertujuan untuk memperkuat hasil tingkat deteksi urin selain berdasarkan warna. Pada bagian input output terdiri dari modul ESP 8266 sebagai pengirim dan penerima data dari server melalui internet.

### 3.2.2 Perancangan User Interface Aplikasi Android

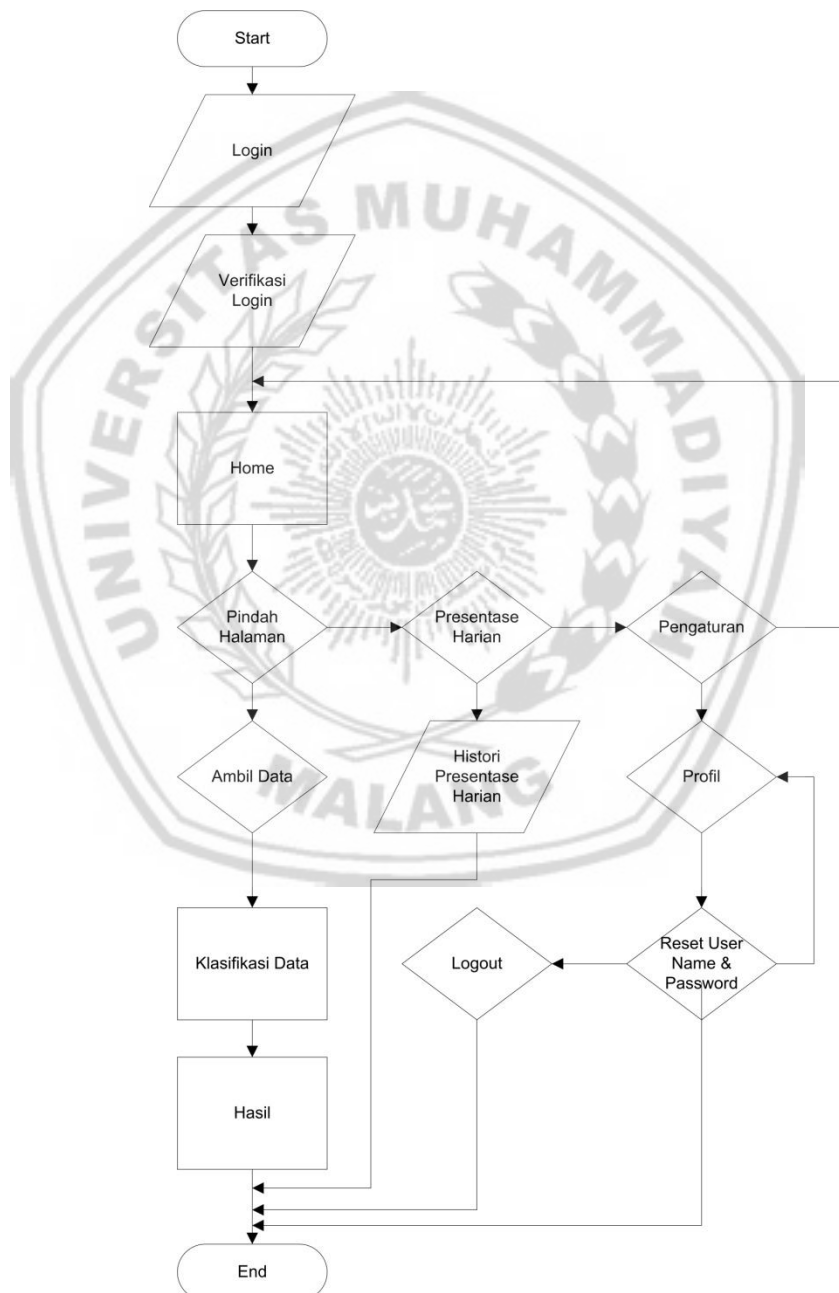
Rancangan user interface aplikasi android sebagai media pemantauan hasil deteksi tingkat dehidrasi terdiri dari halaman login sebagai identifikasi pengguna, halaman aktifitas yang terdiri dari tombol “Ambil Data” dan data presentase harian yang disajikan dalam bentuk angka, halaman history, halaman pengaturan, dan halaman profil.



**Gambar 3.3** Tampilan Halaman Aktifitas

### 3.2.3 Perancangan User Experience Aplikasi Android

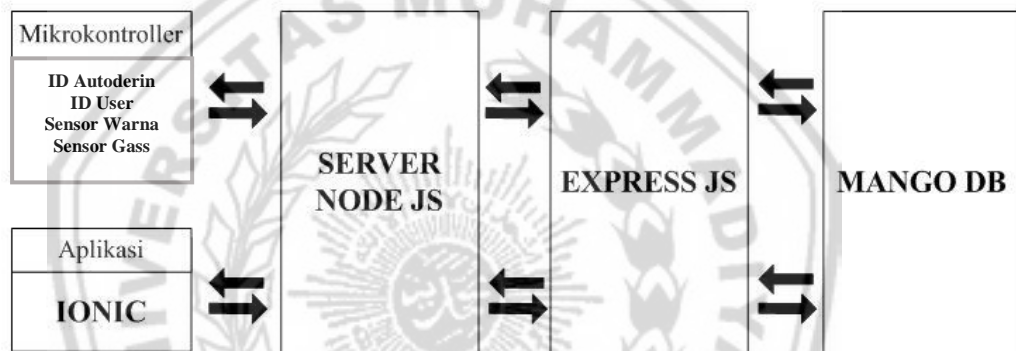
Perancangan user experience sangat diperhatikan dalam pembuatan aplikasi android deteksi dehidrasi tubuh, termasuk sistem komunikasi data antara mikrokontroler, server dan aplikasi android. User experience yang baik akan menghasilkan aplikasi yang interaktif, efisien dan membantu user melakukan sesuatu lebih cepat. Gambar 3.4 berikut adalah flowchart user experience yang akan diterapkan pada aplikasi android deteksi dehidrasi tubuh.



**Gambar 3.4** Flowchart User Experience Aplikasi Android

### 3.2.4 Arsitektur Komunikasi Data

Komunikasi data antara mikrokontroler, server dan aplikasi android yang akan dibangun dengan menggunakan arsitektur Mean Stack. Mean Stack adalah gabungan dari MongoDB, ExpressJS, AngularJS, dan NodeJS, dimana MongoDB sebagai database non-relasional, ExpressJS sebagai framework untuk routing, AngularJS sebagai Javascript framework yang menangani masalah tampilan atau untuk keperluan Front-End Development, dan yang terakhir adalah NodeJS yang menjadi server side untuk Javascript-nya. Pada Gambar 3.5 berikut menampilkan tentang arsitektur komunikasi data antara mikrokontroler, server, dan android.



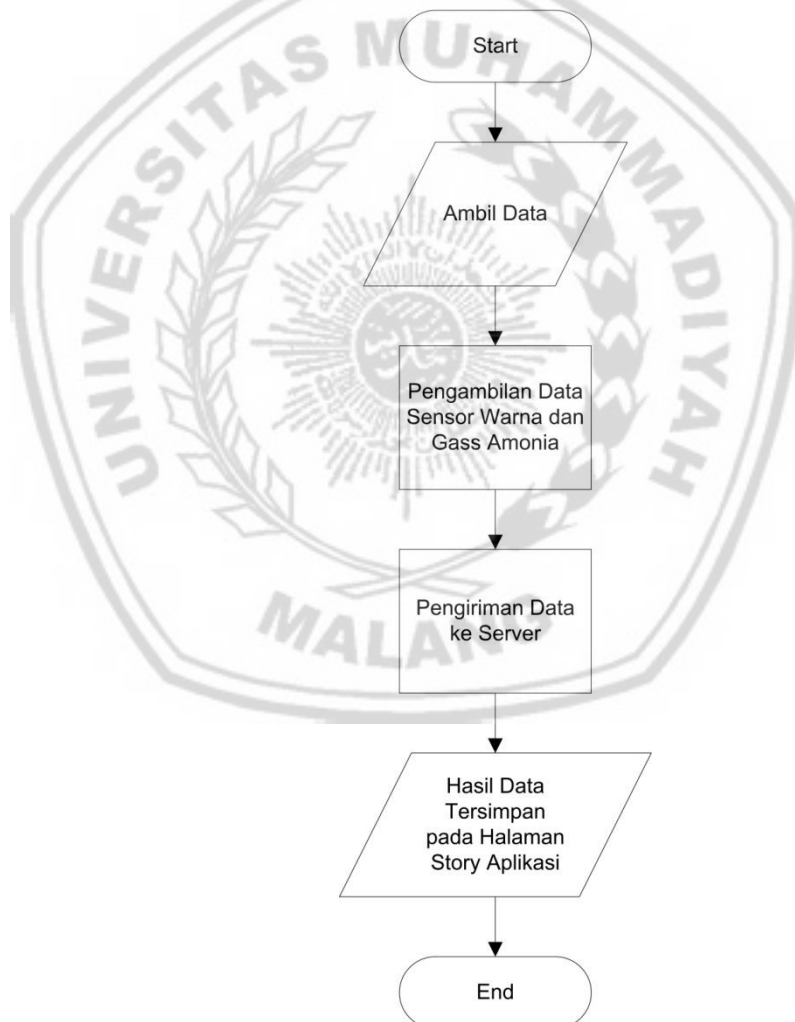
**Gambar 3.5** Perancangan Arsitektur Komunikasi Data

### 3.2.5 Mekanisme Kerja Alat

Mikrokontroller yang digunakan pada alat ini adalah Arduino Uno dengan bahasa pemrograman C++ melalui arduino IDE. Objek utama dalam pengambilan data input yaitu berupa deteksi warna pada urin melalui sensor warna TCS 3200 dan deteksi kadar amonia urin melalui Sensor MQ 135. Data warna dan gass digunakan sebagai parameter untuk menentukan set point sesuai dengan hasil analisa warna yang di kolaborasi dengan klasifikasi menggunakan metode *naïve bayes* yang akan terkirim ke aplikasi android. User dapat mengecek tingkat dehidrasi, saran pengobatan dan history harian tingkat dehidrasi tubuh pada aplikasi android. Ketika user telah membuang air kecil pada wadah Autodein maka sensor warna dan gass akan menginputkan data

pada mikrokontroler untuk memproses klasifikasi tingkat dehidrasi tubuh dan menyimpan data pada mikrokontroler Arduino Uno.

Ketika user membuka aplikasi pada smartphone dan menekan “ Ambil Data “ maka data dari smartphone akan mengirim notifikasi kekontroller melalui perantara modul GSM. Setelah kontroler menerima notifikasi “Ambil Data” maka kontroler akan memberikan feedback pengiriman data hasil klasifikasi deteksi tingkat dehidrasi ke modul GSM dan dari GSM terkirim keaplikasi smartphone dan data dapat dilihat pada aplikasi Seluruh kebutuhan energi listrik dipenuhi oleh catu daya AC to DC 5 volt. Gambar 3.6 merupakan flowchart cara kerja Autoderin secara keseluruhan.



**Gambar 3.6** Flowchart sistem kerja alat

### 3.3 Berdasarkan Warna dan Gass Amonia Urin Terintegrasi Aplikasi Smartphone

Pembuatan perangkat lunak terbagi menjadi 3 bagian, yaitu pembuatan program mikrokontroler, pembuatan software aplikasi android dan pembuatan server.

#### 3.3.1 Penulisan Program Kontroller Arduino uno melalui Arduino IDE

Penulisan logaritma coding arduino uno menggunakan bahasa C+ melalui Arduino IDE sebagai kontroler untuk membaca data sensor warna TCS 3200 dan sensor gass MQ135. Langkah selanjutnya adalah kalibrasi sensor warna dan sensor gass untuk menghasilkan nilai warna dan kandungan gass amonia yang terbaca secara akurat pada Arduino controller. Dan langkah terakhir dalam pembuatan software sistem mekanik yaitu menghubungkan modul GSM, sensor warna, sensor gass dan arduino uno agar data tingkatan dehidrasi dapat diproses melalui handphone seluler. Gambar 3.7 berikut adalah tampilan Arduino IDE untuk menulis program pembacaan sensor.



```
esp | Arduino 1.8.5
File Edit Sketch Tools Help

esp

#include <PubSubClient.h>
#include <ESP8266WiFi.h>
#include <SoftwareSerial.h>
// #include <PubSubClient.h>

WiFiClient wifiConnection;
char* ssid = "Ojo";
const char* password = "weadiganti";

PubSubClient mqttClient(wifiConnection);
const char* mqttHost = "tcp-gateway-1.eu-fr-agera-1.scalingo.com";
const int mqttPort = 20135;

SoftwareSerial toUno(D6, D7);

void setup() {
  Serial.begin(9600);
  while(!Serial){}
  ;
  Serial.println("Serial esp ready");
  toUno.begin(9600);
  Serial.println("Serial to arduino ready");

  delay(10);

  WiFi.mode(WIFI_STA);
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println("\nWiFi connected");
  mqttClient.setServer(mqttHost, mqttPort);
  mqttClient.setCallback(callback);
  mqttClient.connect("arduino1");
  Serial.println("MQTT connected");
}

void loop() {
  mqttClient.loop();
  toUno.write("MQTT connected");
  toUno.flush();
  delay(1000);
}
```

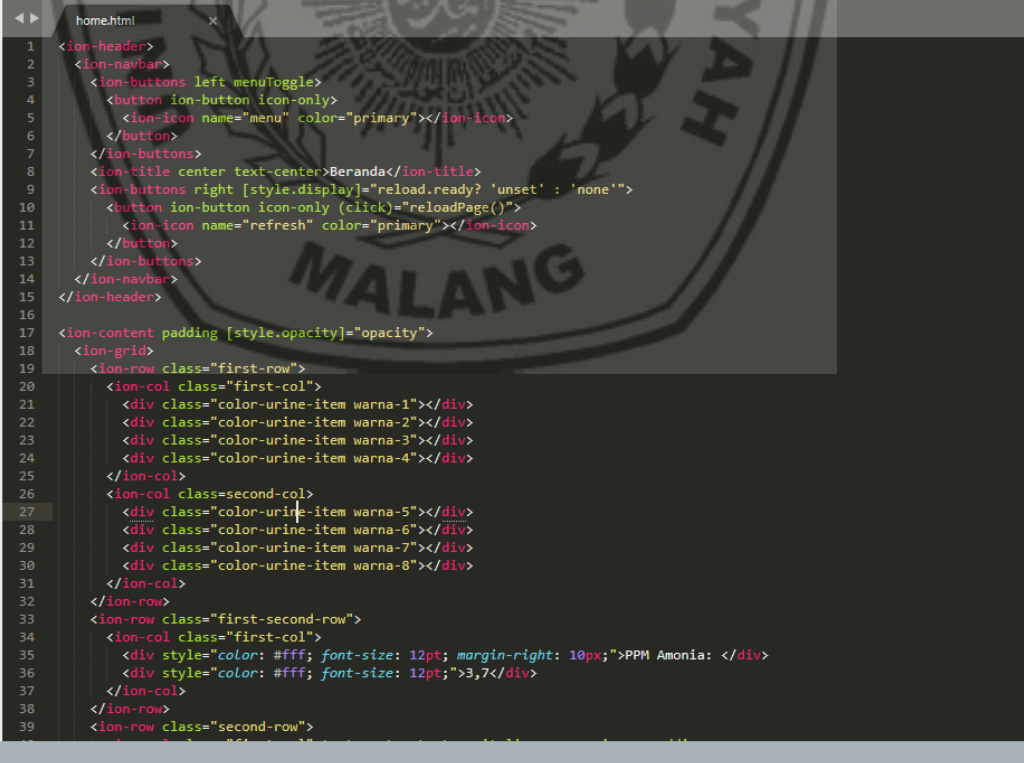
**Gambar 3.7** Penulisan Program Kontroller Melalui Arduino IDE



### 3.3.2 Pembuatan Software Aplikasi Mobile dengan Ionic 3

Pembangunan front end dan back end software aplikasi android untuk mengakses pendeteksi dehidrasi tubuh otomatis berwawasan Internet of Things menggunakan bahasa pemrograman HTML5, CSS 3 dan Javascript dengan framework Ionic 3 dan ExpressJS sebagai framework server side dan dengan aplikasi database MongoDB yang disimpan pada server NodeJS. Dalam pertukaran data dari Cloud database ke lingkungan bahasa Java penulis menggunakan Javascript Object Notation (JSON) sebagai media perantara. Gambar 3.8 adalah pemograman ionic 3 untuk membuat aplikasi mobile.

Untuk membangun aplikasi, beberapa tools dan software dipergunakan. Aplikasi dibangun dan diuji pada platform Windows menggunakan command line interface. Salah satunya adalah penggunaan NodeJS Versi 4.4.0. Untuk menuliskan TypeScript dan CSS Script, editor teks yang digunakan adalah Sublaime Text. Web Browser yang digunakan untuk deployment dan debugging adalah Google Chrome versi 49.0.2623.87.

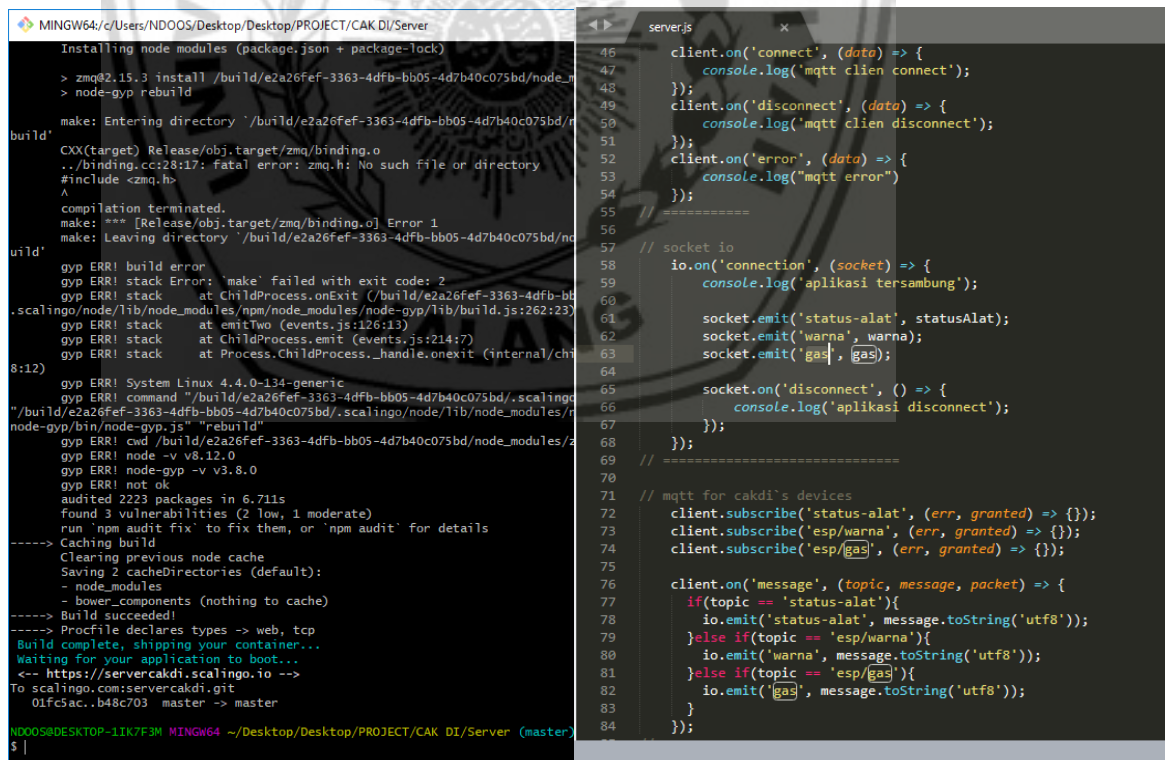


```
home.html
1 <ion-header>
2   <ion-navbar>
3     <ion-buttons left menuToggle>
4       <button ion-button icon-only>
5         <ion-icon name="menu" color="primary"></ion-icon>
6       </button>
7     </ion-buttons>
8     <ion-title center text-center>Beranda</ion-title>
9     <ion-buttons right [style.display]="reload.ready? 'unset' : 'none'">
10       <button ion-button icon-only (click)="reloadPage()">
11         <ion-icon name="refresh" color="primary"></ion-icon>
12       </button>
13     </ion-buttons>
14   </ion-navbar>
15 </ion-header>
16
17 <ion-content padding [style.opacity]="opacity">
18   <ion-grid>
19     <ion-row class="first-row">
20       <ion-col class="first-col">
21         <div class="color-urine-item warna-1"></div>
22         <div class="color-urine-item warna-2"></div>
23         <div class="color-urine-item warna-3"></div>
24         <div class="color-urine-item warna-4"></div>
25       </ion-col>
26       <ion-col class="second-col">
27         <div class="color-urine-item warna-5"></div>
28         <div class="color-urine-item warna-6"></div>
29         <div class="color-urine-item warna-7"></div>
30         <div class="color-urine-item warna-8"></div>
31       </ion-col>
32     </ion-row>
33     <ion-row class="first-second-row">
34       <ion-col class="first-col">
35         <div style="color: #fff; font-size: 12pt; margin-right: 10px;">PPM Amonia: </div>
36         <div style="color: #fff; font-size: 12pt;">3,7</div>
37       </ion-col>
38     </ion-row>
39     <ion-row class="second-row">
```

**Gambar 3.8** Pembuatan Software Aplikasi Android

### 3.3.3 Penyusunan Database dan Pembuatan Server

Penyusunan database sensor warna, sensor gass, dan aplikasi android dibangun dengan menggunakan aplikasi database MongoDB. MongoDB merupakan database open source berbasis dokumen (Document-Oriented Database) yang awalnya dibuat dengan bahasa C++. MongoDB memiliki performa 4 kali lebih cepat dibandingkan MySQL serta mudah diaplikasikan, karena telah tergabung juga sebagai modul PHP. Javascript framework ExpressJS digunakan untuk menangani bagian sisi server atau Back End aplikasi seperti routing dan manajemen session. Sedangkan untuk input dan output servernya sendiri menggunakan NodeJS. NodeJS merupakan platform untuk membangun Real-time Application. NodeJS dapat menangani event input-output server, dengan kata lain NodeJS dapat memungkinkan para developer Javascripts untuk membuat event-driven servers dalam JavaScript. Dengan menggunakan NodeJS, pemakaian memori lebih hemat. Gambar 3.9 adalah konfigurasi database dan server.



```
MINGW64~/c:/Users/ND00S/Desktop/Desktop/PROJECT/CAK DI/Server
Installing node modules (package.json + package-lock)
> zmq@2.15.3 install /build/e2a26fef-3363-4dfb-bb05-4d7b40c075bd/node_modules/zmq
> node-gyp rebuild

make: Entering directory '/build/e2a26fef-3363-4dfb-bb05-4d7b40c075bd/node_modules/zmq'
CXX(target) Release/obj.target/zmq/binding.o
../binding.cc:28:17: fatal error: zmq.h: No such file or directory
#include <zmq.h>
^
compilation terminated.
make: *** [Release/obj.target/zmq/binding.o] Error 1
make: Leaving directory '/build/e2a26fef-3363-4dfb-bb05-4d7b40c075bd/node_modules/zmq'

gyp ERR! build error
gyp ERR! stack Error: 'make' failed with exit code: 2
gyp ERR! stack at ChildProcess.onExit (/build/e2a26fef-3363-4dfb-bb05-4d7b40c075bd/node_modules/node-gyp/lib/build.js:262:23)
gyp ERR! stack at emitTwo (events.js:126:13)
gyp ERR! stack at ChildProcess.emit (events.js:214:7)
gyp ERR! stack at Process.ChildProcess._handle.onexit (internal/child_process.js:178:12)
gyp ERR! System Linux 4.4.0-134-generic
gyp ERR! command "/build/e2a26fef-3363-4dfb-bb05-4d7b40c075bd/node_modules/node-gyp/bin/node-gyp.js" "rebuild"
gyp ERR! cwd /build/e2a26fef-3363-4dfb-bb05-4d7b40c075bd/node_modules/zmq
gyp ERR! node -v v8.12.0
gyp ERR! node-gyp -v v3.8.0
gyp ERR! not ok
audited 2223 packages in 6.711s
found 3 vulnerabilities (2 low, 1 moderate)
run 'npm audit fix' to fix them, or 'npm audit' for details
-----
Caching build
Clearing previous node cache
Saving 2 cacheDirectories (default):
- node_modules
- bower_components (nothing to cache)
-----
Build succeeded!
-----
Procfile declares types -> web, tcp
Build complete, shipping your container...
Waiting for your application to boot...
<-- https://servercakdi.scalingo.io -->
To scalingo.com:servercakdi.git
01fc5ac..b48c703 master -> master

ND00S@DESKTOP-11K7F3M MINGW64 ~/Desktop/Desktop/PROJECT/CAK DI/Server (master)
$ |
```

```
server.js
46 client.on('connect', (data) => {
47   console.log('mqtt clien connect');
48 });
49 client.on('disconnect', (data) => {
50   console.log('mqtt clien disconnect');
51 });
52 client.on('error', (data) => {
53   console.log("mqtt error");
54 });
55 // =====
56 // socket io
57 io.on('connection', (socket) => {
58   console.log('aplikasi tersambung');
59
60   socket.emit('status-alat', statusAlat);
61   socket.emit('warna', warna);
62   socket.emit('gas', gas);
63
64   socket.on('disconnect', () => {
65     console.log('aplikasi disconnect');
66   });
67 });
68 // =====
69 // mqtt for cakdi's devices
70
71 client.subscribe('status-alat', (err, granted) => {});
72 client.subscribe('esp/warna', (err, granted) => {});
73 client.subscribe('esp/gas', (err, granted) => {});
74
75 client.on('message', (topic, message, packet) => {
76   if(topic === 'status-alat'){
77     io.emit('status-alat', message.toString('utf8'));
78   }else if(topic === 'esp/warna'){
79     io.emit('warna', message.toString('utf8'));
80   }else if(topic === 'esp/gas'){
81     io.emit('gas', message.toString('utf8'));
82   }
83 });
84
```

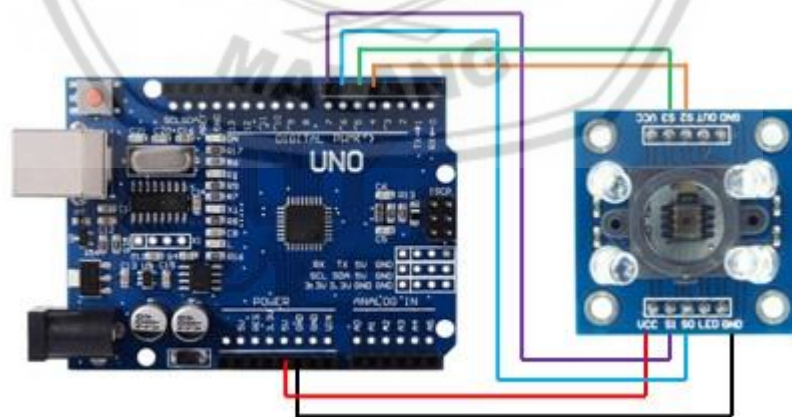
Gambar 3.9 Konfigurasi Database dan Server

### 3.3.4 Perancangan Hardware Pendeteksi Tingkat Dehidrasi Tubuh Berdasarkan Warna dan Kadar Amonia Urin yang Terintegrasi Aplikasi Smartphone

Perancangan Hardware dilakukan secara bertahap mulai dari Perancangan Sensor warna TCS 3200, sensor gass MQ 135, dan modul wifi esp8266.

### 3.3.5 Perancangan Sensor Warna TCS 3200

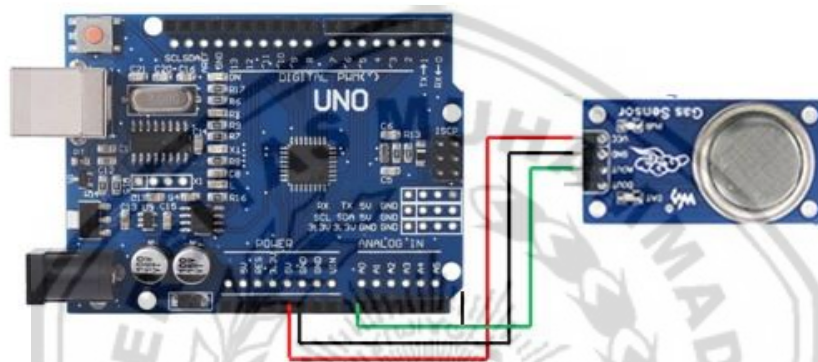
Perancangan Sensor warna pada sistem ini digunakan untuk mengetahui tingkatan warna dalam katalog tingkat dehidrasi tubuh berdasarkan warna urin. Sensor warna yang digunakan dalam penelitian ini adalah sensor warna TCS 3200. Sensor ini memiliki 6 buah kabel yaitu merah (power suplay +), hitam (ground -), biru, ungu, oranye, dan hijau merupakan digital PWM. Rangkaian Sensor Warna TCS 3200 dihubungkan dengan Arduino dengan kabel merah ke VCC, kabel hitam pada GND, kabel ungu pada pin S1, kabel biru pada S0, kabel hijau pada pin chanel E dan 2 kabel data pada chanel A, dan dari modul HX711 mempunyai 4 kabel yang masing – masing dihubungkan pada 5v power suply, gnd dan 2 pin input analaog arduino uno seperti yang ditunjukkan pada gambar 3.10



**Gambar 3.10** Rangkaian Sensor Warna TCS 3200 pada Arduino Uno

### 3.3.6 Perancangan Sensor Gass MQ 135

Perancangan sensor gass pada sistem ini digunakan untuk mengetahui kadar amonia pada cairan urin. Sensor secara otomatis akan membaca data ketika mendeteksi kandungan gass amonia disekitar wadah tempat cairan urin. Sensor gass ini mempunyai 3 pin koneksi yang akan dihubungkan pada arduino uno yaitu data analog A0, Vcc, dan Gnd. Pin data mengeluarkan output tegangan TPP aroma gass amonia urin yang akan di baca oleh arduino uno. Berikut adalah konfigurasi antara sensor gass dengan pin arduino uno.

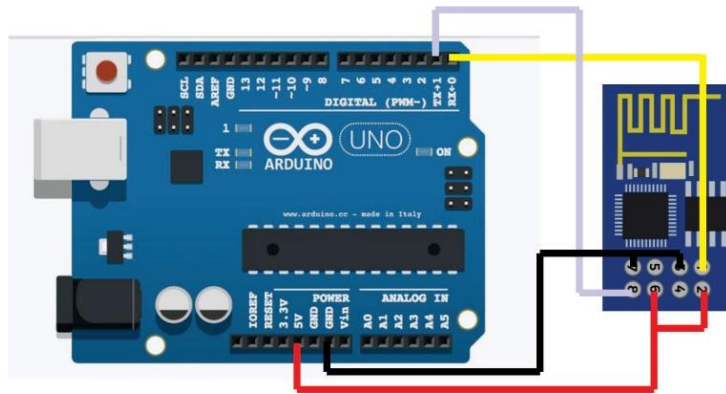


**Gambar 3.11** Perancangan Rangkaian Sensor Gass MQ 135

### 3.3.7 Perancangan Rangkaian Modul Wi-Fi ESP8266

Modul wireless ESP8266 merupakan modul low-cost Wi-Fi dengan dukungan penuh untuk penggunaan TCP/IP. Salah satu seri ESP8266 adalah seri ESP-01. Modul Esp8266 berfungsi untuk mengoneksikan perangkat Autoderin dengan internet dan menjadi jembatan komunikasi yang nantinya akan mengirim maupun menerima perintah dari ke Arduino melalui perantara wifi. Modul esp8266 ini mempunyai beberapa pin, pin tx, rx, vcc, gnd dan ch\_pd saja yang dipakai jika modul esp8266 hanya digunakan sebagai modul wifi. Pin tx dihubungkan dengan pin tx arduino, pin rx dihubungkan pin rx arduino, vcc dan ch\_pd dihubungkan pada pin 3,3 power suply arduino dan gnd dihubungkan pada pin gnd arduino uno seperti yang ditunjukkan pada gambar 3.12





**Gambar 3.12** Rangkaian Modul ESP8266 pada Arduino Uno

### 3.3.8 Perancangan Server Dedicated

Spesifikasi minimum server yang dibutuhkan untuk komunikasi data antara perangkat Autoderin dengan aplikasi smartphone adalah Processor AMD Turion(tm) II P540 Dual-Core Processor (2 CPUs), ~2.4GHz, memory RAM 2 GB, network card, Hard disk , Minimum 80 GB 5400 rpm, dan Motherboard Msi 941+hsf.

### 3.4 Metode Pengujian

Tahapan pengujian ini bertujuan untuk menentukan cara pengujian alat yang akan dilakukan pada pendeteksi tingkat dehidrasi tubuh berdasarkan warna dan kadar amonia urin terintegrasi aplikasi smartphone. Ada beberapa pengujian antara lain:

1. Pengujian Mikrokontroller
2. Pengujian ESP 8266
3. Pengujian user experience pada aplikasi Android
4. Pengujian Sensor warna TCS 3200
5. Pengujian Sensor gass MQ 135
6. Pengujian sistem monitoring pada aplikasi Android
7. Pengujian seluruh sistem Autoderin.

### 3.4.1 Pengujian Mikrokontroler

#### 3.4.1.1 Tujuan

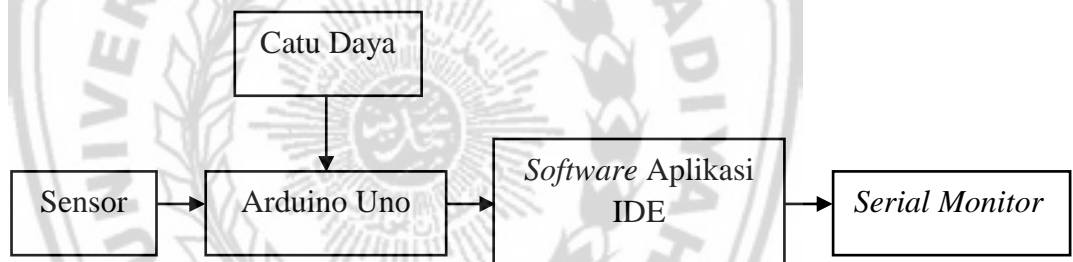
Pengujian dilakukan untuk mengetahui apakah mikrokontroler arduino uno dapat bekerja dalam kondisi baik atau tidak dan dapat mengkonversi nilai tegangan menjadi sebuah data sensor.

#### 3.4.1.2 Peralatan

Untuk melakukan pengujian tersebut diperlukan perlengkapan berupa:

1. Rangkaian Catu daya
2. Rangkaian mikrokontroller dan sensor
3. Kabel konektor
4. *Software* IDE Arduino

#### 3.4.1.3 Blok Diagram



**Gambar 3.18** Diagram Blok Pengujian Mikrikontroler

#### 3.4.1.4 Persiapan

1. Menghubungkan sensor pada pin mikrokontroller
2. Membuat listing program pada *software* aplikasi IDE agar Sensor warna, dan sensor gas dapat ditampilkan pada serial monitor.
3. Mengunggah program pada mikrokontroler arduino uno.
4. Membuka serial monitor.

### 3.4.2 Pengujian ESP 8266

#### 3.4.2.1 Tujuan

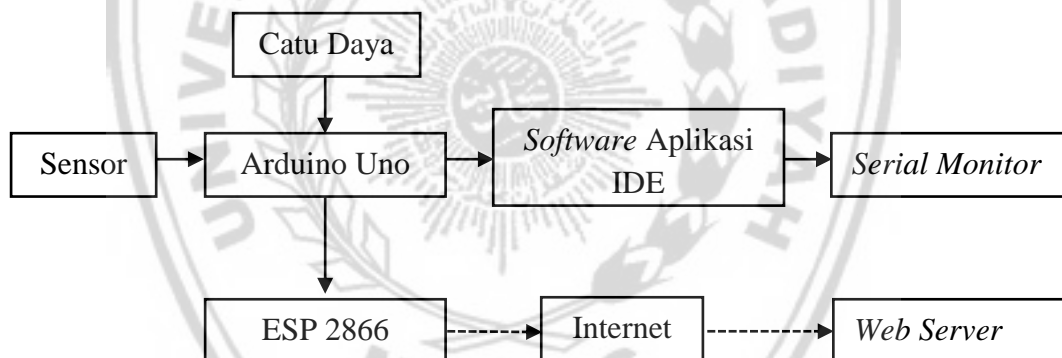
Pengujian dilakukan untuk mengetahui apakah esp 8266 dapat bekerja dalam kondisi baik dan dapat mengirim data dari mikrokontroler pada server.

#### 3.4.2.2 Peralatan

Untuk melakukan pengujian tersebut diperlukan perlengkapan berupa:

1. Rangkaian Catu daya
2. Rangkaian mikrokontroller dan sensor
3. Modul ESP 8266
4. Kabel konektor
5. *Software* IDE Arduino
6. Internet
7. Web Server

#### 3.4.2.3 Blok Diagram



**Gambar 3.19** Diagram Blok Pengujian ESP 2866

#### 3.4.2.4 Persiapan

1. Menghubungkan sensor pada pin mikrokontroller
2. Membuat listing program pada *software* aplikasi IDE agar Sensor warna dan Sensor gass dapat ditampilkan pada serial monitor.
3. Mengunggah program pada mikrokontroler arduino uno.
4. Membuka serial monitor.
5. Membuat listing program pada *software* aplikasi IDE untuk pengambilan data sensor pada arduino dan mengatur pengiriman data pada alamat server.

6. Membuka web server.

### 3.4.3 Pengujian User Experience Aplikasi Android

#### 3.4.3.1 Tujuan

Pengujian dilakukan untuk menganalisa sistem kerja aplikasi pendeteksi tingkat dehidrasi tubuh apakah menghasilkan aplikasi yang interaktif, efisien dan membantu user melakukan sesuatu lebih cepat yang fungsinya sebagai media informasi mengenai tingkat dehidrasi tubuh berdasarkan warna dan kadar amonia pada urin. Pengecekan dilakukan dengan menekan tombol “Ambil Data”.

#### 3.4.3.2 Persiapan

Untuk pengujian user experience aplikasi android otomasi pemanen madu diperlukan langkah – langkah berikut :

1. Menginstal aplikasi android pendeteksi tingkat dehidrasi tubuh pada *smartphone* android minimal pada *OS android 4.4*
2. Setelah aplikasi selesai di instal, ikon aplikasi akan muncul pada *screen home* pada *smartphone* dan klik ikon untuk membuka aplikasi.
3. Setelah aplikasi terbuka maka akan muncul halaman login terdiri dari kolom email, *password* dan daftar. Klik ikon daftar untuk membuat akun untuk melanjutkan halaman berikutnya.
4. Setelah data pada halaman *login* terdaftar oleh server maka akan tampil halaman Beranda yang terdapat data informasi konektifitas server, warna tingkat dehidrasi dan kadar kandungan gass urin dan terdapat tombol “Ambil Data” yang jika ditekan maka akan mendownload data dari kontroler mengenai deteksi tingkat dehidrasi tubuh.
5. Klik ikon *side menu* pada kiri atas untuk memilih halaman, terdapat halaman beranda, history, logout dan ikon edit profil.
6. Pengguna dapat memilih halaman history untuk melihat hasil deteksi tingkat dehidrasi sebelumnya dalam bentuk tabel.
7. Pengguna dapat memilih ikon edit profil untuk mengedit data untuk keperluan *login* dan mengganti foto profil.



8. Pengguna dapat memilih ikon *logout* untuk keluar dari aplikasi.

